

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Jan Terzer

Topološki model za brezžična senzorska omrežja

DIPLOMSKO DELO

UNIVERZITETNI ŠTUDIJSKI PROGRAM PRVE
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

Ljubljana, 2016

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Jan Terzer

Topološki model za brezžična senzorska omrežja

DIPLOMSKO DELO

UNIVERZITETNI ŠTUDIJSKI PROGRAM PRVE
STOPNJE RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: prof. dr. Nežka Mramor Kosta

Ljubljana, 2016

To delo je ponujeno pod licenco *Creative Commons Priznanje avtorstva-Deljenje pod enakimi pogoji 2.5 Slovenija* (ali novejšo različico). To pomeni, da se tako besedilo, slike, grafi in druge sestavine dela kot tudi rezultati diplomskega dela lahko prosto distribuirajo, reproducirajo, uporabljajo, priobčujejo javnosti in predelujejo, pod pogojem, da se jasno in vidno navede avtorja in naslov tega dela in da se v primeru spremembe, preoblikovanja ali uporabe tega dela v svojem delu, lahko distribuira predelava le pod licenco, ki je enaka tej. Podrobnosti licence so dostopne na spletni strani creativecommons.si ali na Inštitutu za intelektualno lastnino, Streliška 1, 1000 Ljubljana.



Izvorna koda diplomskega dela, njeni rezultati in v ta namen razvita programska oprema je ponujena pod licenco *GNU General Public License*, različica 3 (ali novejša). To pomeni, da se lahko prosto distribuira in/ali predeluje pod njenimi pogoji. Podrobnosti licence so dostopne na spletni strani <http://www.gnu.org/licenses>.¹

Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Tematika naloge:

Brezžična senzorska omrežja, ki zajemajo različne podatke na svojem območju, so čedalje bolj pogost vir informacij o našem okolju. Pogosto natančne lokacije posameznih senzorjev niso znane, poznamo le njihov komunikacijski domet. Takšna senzorska omrežja lahko uspešno modeliramo s pomočjo simplicialnih kompleksov. Topološke lastnosti kot so homološke grupe in Bettijeva števila simplicialnega kompleksa zgrajenega na senzorskem omrežju povedo, ali je območje senzorskega omrežja pokrito ali pa so v njem kakšne luknje, ki jih senzorji omrežja ne dosežejo.

V diplomskem delu opišite topološki model brezžičnega senzorskega omrežja in izdelajte simulator za prikaz njegovega območja pokritosti. Simulator naj nad senzorskih omrežjem iz podatkov o dosegu senzorjev zgradi Vietoris-Ripsov kompleks, izračuna prvo Bettijevo število in vrne podatke o pokritosti območja.

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani Jan Terzer sem avtor diplomskega dela z naslovom:

Topološki model za brezžična senzorska omrežja

(angl. A Topological model for wireless sensor networks)

S svojim podpisom zagotavljam, da:

- sem diplomsko delo izdelal samostojno pod mentorstvom prof. dr. Nežka Mramor Kosta,
- so elektronska oblika diplomskega dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko diplomskega dela,
- soglašam z javno objavo elektronske oblike diplomskega dela na svetovnem spletu preko univerzitetnega spletnega arhiva.

V Ljubljani, dne 4. februar 2016

Podpis avtorja:

*Zahvaljujem se mentorici prof. dr. Nežka Mramor Kosta za pomoč in podporo pri
pisanju diplomskega dela.*

Kazalo

Povzetek

Abstract

Poglavje 1	Uvod	1
1.1	Senzorji in njihove zaznavne sposobnosti	2
1.2	Predpostavke	3
Poglavje 2	Orodja računske topologije	4
2.1	Topologija	4
2.2	Simplicialni kompleksi	4
2.3	Vietoris-Ripsov kompleks	7
2.3.1	Opomba	9
2.4	Homologija	10
2.4.1	Grupe verig	10
2.4.2	Cikli in robovi	12
2.4.3	Homološke grupe	14
2.5	Postopek računanja	15
Poglavje 3	Simulator	18
3.1	Opis	18
3.2	Grafična predstavitev	22
3.3	Funkcionalnost	23
Poglavje 4	Zaključek	27
Literatura		28

Povzetek

Naslov: Topološki model za brezžična senzorska omrežja

Senzorji so naprave za pridobivanje informacij iz okolja. Da pokrijemo večjo površino zaznavanja, jih razporedimo po željeni domeni in med seboj povežemo v senzorsko omrežje. Postavi se vprašanje ali senzorji pokrivajo celotno površino domene, oz. ali obstaja območje znotraj domene, ki ga senzorji ne dosežejo. Taka območja imenujemo luknje. V brezžičnem senzorskem omrežju s senzorji, ki se lahko premikajo, poznamo samo njihovo medsebojno povezanost, ne pa tudi položaja. Pokritost območja lahko v takšnih primerih ugotovimo s topološkimi metodami, tako da iz podatkov o povezanosti senzorjev zgradimo Ripsov kompleks in izračunamo njegove homološke grupe oziroma Bettijeva števila. V diplomski nalogi je predstavljen simulator za generiranje senzorskih omrežij in za izračun števila lukenj v domeni pokritosti na podlagi izračuna prvega Bettijevega števila ustreznega Ripsovega kompleksa.

Ključne besede: Bettijeva števila, homološke grupe, Ripsov kompleks, simplicialni kompleks, brezžična senzorska omrežja, območje pokritosti

Abstract

Title: Topological model for wireless sensor networks

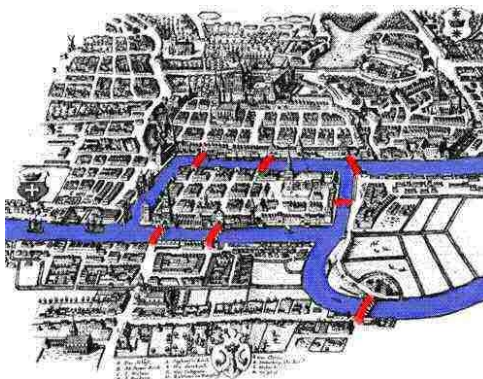
Sensors are devices for extracting information from the environment. In order to cover a larger area of detection, they are spread across the desired domain and connected in a sensor network. The question arises whether the sensors cover the entire area of domain or in other words whether there is an area within the domain that sensors do not detect. An area such as this is called a hole. In wireless non-stationary sensor networks the location of the individual sensors is not known, only information on which sensors are connected is available. The coverage of the domain of such a sensor network can be determined using topological methods. Using information on connectivity between the sensors we build the Rips complex and compute its homology groups and Betti numbers. The thesis presents a simulator for generating sensor networks and calculating the number of holes in it by computing the first Betti number of the corresponding Rips complex.

Keywords: Betti numbers, homology groups, Rips complex, simplicial complex, wireless sensor networks, domain coverage

Poglavje 1 Uvod

Senzorska omrežja postajajo vedno bolj pomemben in prodoren del sodobnega računanja in avtomatizacije. V tako aktivnem in hitro razvijajočem se področju imamo pogosto težave s pokritostjo območja. Problem iskanja lukenj v senzorskem omrežju se pojavlja na različnih področjih: zaznavanje okolja, komunikacija in oddajanje, robotska navigacija, nadzorovanje, varnost in vojaški nameni. Za lažjo predstavo, si kot primer, vzemimo nabor vozlišč X , na robu domene D na ravnini. Predpostavimo, da lahko vsako vozlišče zaznava in pošilja signal, ali pa kako drugače pokriva okolico znotraj kroga z nekim polmerom pokritosti okoli sebe. Osnovno vprašanje, ki se nam postavi glede pokritosti območja, je: »Ali nabor vozlišč X pokrije celotno območje D , torej ali množica krogov s središči v vozliščih X in s podanimi polmeri pokritosti, vsebuje domeno D ?«

Področje topologije se že vrsto let ukvarja z iskanjem odgovorov na takšna vprašanja. Začetki topologije segajo v leto 1736, ko je Euler prvič objavil nerešljiv problem Königsbergovih mostov (Slika 1) v članku z naslovom *Solutio problematis ad geometriam situs pertinentis* [3], kar pomeni *Rešitev problema v povezavi z geometrijsko pozicijo* [6]. Do konca prejšnjega stoletja je bila topologija eno od teoretičnih področij matematike, v zadnjih letih pa so se topološke metode uveljavile tudi kot primerne pri obravnavanju bolj praktičnih problemov [4],[1]. Eden od njih je analiza pokritosti senzorskih omrežij [10],[9]. Z uporabo cenovno ugodnih senzorjev, ki niso večji od kovanca, se je dopustno število senzorjev v senzorskem omrežju močno povečalo. Namesto nekaj posameznih senzorjev, se za pridobivanje informacij iz okolja uporablja na tisoče manjših.



Slika 1: Problem Königsbergovih mostov [6]

Podobno kot Euler se srečujemo z geometrijskim problemom, kjer razdalja med senzorji ni pomembna oz. ni znana. Znano je le, kako so senzorji med seboj povezani. Želimo pa ugotoviti, ali v celoti pokrijejo območje.

V diplomskem delu bomo predstavili in opisali simulator, ki smo ga sprogramirali za generiranje senzorskega omrežja in izračun njegove pokritosti s topološkimi metodami. Podani bodo zadostni kriteriji za dosego pokritosti. Vozlišča bomo razporedili vnaprej poljubno, ne pa po nekem fiksnem protokolu oz. vzorcu. S porazdelitvijo in gostoto vozlišč se ne bomo obremenjevali. Kriterij pokritosti, ki bo predstavljen, je izračunljiv in centraliziran. Za osnovo diplomskega dela smo vzeli članek [8], v katerem je razložen koncept odkrivanja lukenj s topološkimi metodami v domeni z znanimi robnimi senzorji. Avtorji članka so izbrali pristop, ki zahteva poznavanje podatkov o tem, kateri senzorji so na robu domene, za katero preverjamo pokritost. Naš pristop je nekoliko drugačen, saj pri računanju pokritosti domene ne zahtevamo poznavanja mejnih senzorjev, temveč le podatke o medsebojni povezanosti vseh senzorjev. V preostanku tega poglavja bomo opisali predpostavke, ki veljajo za senzorska omrežja. V naslednjem poglavju so opisani topološki pojmi, ki jih bomo uporabili: simplicialni kompleksi, Ripsov kompleks, homološke grupe ter Bettijeva števila. Tretje poglavje vsebuje opis delovanja simulatorja in programa. Zadnje poglavje pa povzema rezultate diplomskega dela.

1.1 Senzorji in njihove zaznavne sposobnosti

Senzor je naprava, ki meri določene lastnosti domene ali okolja kjer se nahaja in vrača signal, na podlagi katerega lahko dobimo željene informacije. Te informacije so lahko na primer temperatura, svetlost in vlažnost vsakega prostora v pametnih hišah, na podlagi katerih lahko računalnik pravilno reagira glede na želje uporabnika; poveča gretje, odpre okna, prižge luči itd. Senzorji se razlikujejo po obsegu, ločljivosti in zmožnostih. Informacije, ki jih posreduje, so lahko preproste binarne zastave (spremenljivka, ki ima vrednosti 0 ali 1). Podobno je pri napravi za zaznavanje kovine, ki s piskom naznani njeno detekcijo. Obstajajo pa tudi bolj kompleksni senzorji, kot so video kamere. Te pošiljajo signal, ki potrebuje podrobnejšo analizo, za pridobitev željenih podatkov [7].

Vedno bolj pogosto se uporabljajo senzorji za pregledovanje določenih objektov ali snovi v regijah. Ustreden primer je zaznavanje in lociranje izbruha požara v gozdovih, zaznavanje radioloških in bioloških nevarnosti, skritih min ter streliva, ali pa celo posamezne osebe v množici ljudi. Vsi naštet primeri so glede podatkovnega urejanja zahtevni in težavni problemi.

1.2 Predpostavke

V diplomskem delu bomo obravnavali senzorska omrežja, kjer natančni položaji posameznih senzorjev niso znani. Edini podatek, ki ga uporabljamo za analizo pokritosti omrežja, je povezanost med posameznimi vozlišči kjer se nahajajo senzori. Osnovne predpostavke našega modela so:

A1: Vozlišča X oddajajo unikatno ID številko. Vsako vozlišče lahko zazna identiteto vsakega drugega vozlišča, ki je v njegovem **dosegu oddajanja** r_b .

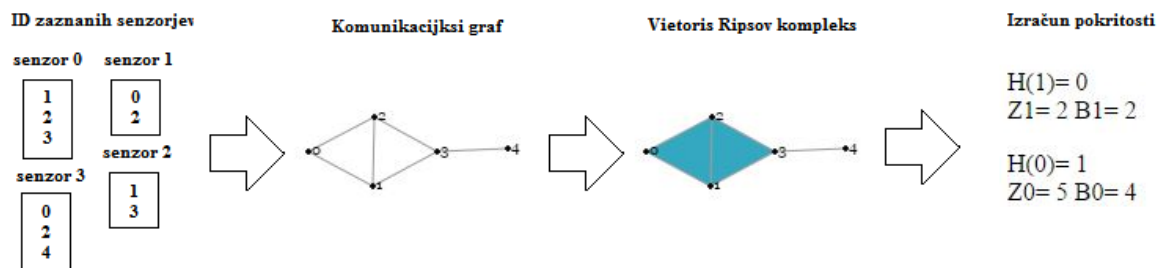
A2: Vozlišča imajo radialno enakomerno domeno pokritosti znotraj **dosega pokritosti** $r_c \geq r_b/\sqrt{3}$.

A3: Domena D leži na ravnini; pri računanju pokritosti ne upoštevamo višine senzorjev.

Dosega r_b in r_c sta za vse senzore v omrežju enaka. Podatek, ki ga vsebuje vsako vozlišče omrežja, je seznam ID številc vozlišč, ki so v dosegu njegovega signala, torej so oddaljena od vozlišča za manj kot r_b .

Naloga, ki jo bomo reševali je, da ob danih predpostavkah ugotovimo, ali je območje senzorskega omrežja v celoti pokrito, ali pa so v njem luknje, torej predeli, ki jih omrežje ne pokriva.

Nekoliko presenetljivo je, da ti grobi podatki v veliko primerih zadoščajo, da natančno določimo pokritost omrežja. Iz njih lahko zgradimo komunikacijski graf, katerega točke so vozlišča omrežja, povezave pa predstavljajo komunikacijsko povezljivost med vozlišči (z radijem r_b). Iz grafa zgradimo simplicialni kompleks, ki ga imenujemo Ripsov kompleks \mathcal{R} ; največji simplicialni kompleks z istimi vozlišči in povezavami. Iz homoloških grup Ripsovega kompleksa in njihovih rangov, imenovanih Bettijeva števila, pa lahko ugotovimo ali je področje pokrito in identificiramo luknje. Slika 2 je shematični prikaz postopka računanja pokritosti domene preprostega senzorskega omrežja.



Slika 2: Postopek izračuna pokritosti

Poglavje 2 Orodja računske topologije

2.1 Topologija

Topologija obravnava lastnosti množic ali objektov, ki se ohranjajo skozi deformacije kot so raztezanje in upogibanje, ne pa tudi trganje in lepljenje. Tako je krog, če ga razegnemo, s stališča topologije enak elipsi in kroglu elipsoidu. S strani topologije je pomembno, koliko lukenj ima opazovano telo in ne sama oblika telesa. Tako je krog topološko različen od lika v obliki številke 8 in krogla različna od telesa v obliki votlega obroča ali torusa.

V senzorskih omrežjih pogosto ne moremo dobiti informacije o lokaciji senzorjev, ne poznamo njihove medsebojne razdalje in smeri iz katere prihaja signal. Zato je odsotnost lokalizacije za naš problem pomembna in potrebna lastnost topologije. Vse potrebne informacije za izračun pokritosti topološkega prostora so povezave med vozlišči in vozlišča sama. Torej zanima nas samo, ali obstaja povezava med dvema vozliščema, ne pa tudi njuna neposredna lega. V nadaljevanju bomo za lažje razumevanje potrebovali poznavanje naslednjih pojmov: simplicialni kompleksi, Vietoris Rips konstrukcija, homološke grupe in Bettijeva števila. Opise teh pojmov povzemamo po [2],[8],[13].

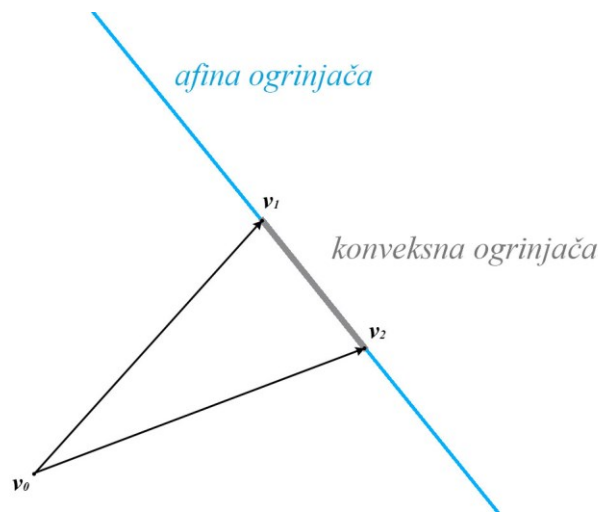
2.2 Simplicialni kompleksi

Simplicialni kompleks je najpogostejši matematični model, ki se uporablja v topološki analizi podatkov. Uporabljamo ga za rekonstrukcijo matematičnega modela originala iz oblaka množice točk, ki jih povežemo s preprostimi geometrijskimi elementi kot so: daljice, trikotniki, tetraedri, itd. Vsi ti elementi so simpleksi različnih dimenzij, da zadostijo pogoju simplicialnega kompleksa, pa morajo biti pravilno sestavljeni.

Množica točk $\{v_0, \dots, v_k\}$ iz prostora \mathbb{R}^d je *afino neodvisna*, če so vektorji $v_i - v_0$, kjer je $1 \leq i \leq k$, linearno neodvisni.

Pri definiciji ni pomembno kateri vektor je zapisan prvi, izberemo lahko poljubnega. Lastnost afine neodvisnosti se prenese tudi na podmnožice, vsaka podmnožica afino neodvisnih točk je afino neodvisna.

Vzemimo, da so točke v_0, v_1, \dots, v_k v prostoru \mathbb{R}^d afino neodvisne. V prostoru dimenzije d imamo lahko največ d linearno neodvisnih vektorjev in zato tudi največ $d + 1$ afino neodvisnih točk. Točka $x = \sum_{i=0}^k \lambda_i v_i$ je *afina kombinacija* vektorjev v_i , če je $\sum_{i=0}^k \lambda_i = 1$ (Slika 3).



Slika 3: Afina in konveksna ogrinjača

Množico vseh afinih kombinacij točk iz množice $V = \{v_0, v_1, \dots, v_k\}$ imenujemo *afina ogrinjača* množice V . Afina ogrinjača dimenzije 0 je točka, dimenzije 1 premica, dimenzije 2 ravnina, višjih dimenzij pa afini podprostor. Afino kombinacijo $\sum_{i=0}^k \lambda_i v_i$, kjer so vsi koeficienti λ_i pozitivni, imenujemo *konveksna kombinacija*, množico vseh konveksnih kombinacij dane množice točk V pa *konveksna ogrinjača* množice V .

Simpleks dimenzije k ali *k-simpleks* σ je konveksna ogrinjača $k + 1$ afino neodvisnih točk $\text{conv}\{v_0, v_1, \dots, v_k\}$, ki jim pravimo *oglišča* simpleksa. Simpleksi nižjih dimenzij imajo tudi posebna imena: 0-simpleks imenujemo točka, 1-simpleks daljica, 2-simpleks trikotnik in 3-simpleks tetraeder. Lastnost afine neodvisnosti se prenaša na podmnožice, zato je tudi konveksna ogrinjača podmnožice oglišč danega simpleksa nov simpleks.

Lice simpleksa σ je konveksna ogrinjača neprazne podmnožice njegovh oglišč. Če podmnožica ni celotna množica točk v_i , potem to ogrinjačo imenujemo pravo lice. Če je τ lice simpleksa σ , potem rečemo, da je σ kolice simpleksa τ in to zapišemo kot $\tau \leq \sigma$ ali pa $\tau < \sigma$, če τ je pravo lice. Množica velikosti $k + 1$ ima 2^{k+1} podmnožic, med katere štejemo tudi prazno množico. Simpleks σ dimenzije k ima torej $2^{k+1} - 1$ lic, od katerih so vsa razen enega prava.

Za nas je pojem lice pomemben, ker z njim definiramo rob simpleksa. Za primer lahko vzamemo poln trikotnik. Lica trikotnika so vse njegove stranice in oglišča, njihova unija pa predstavlja rob trikotnika. Rob σ , ki ga označimo z $\text{bd } \sigma$, je torej unija vseh pravih lic. Vse ostalo je njegova notranjost, int $\sigma = \sigma - \text{bd } \sigma$. Točka $x = \sum \lambda_i v_i \in \sigma$ spada v notranjost σ samo, če so vsi koeficienti λ_i večji od 0. Takoj, ko ima eden od koeficientov vrednost nič pomeni, da točka x pade na rob simpleksa. To je tudi način kako računsko določimo vsa prava lica. Najprej postavimo vrednost koeficienta samo enega oglišča na 0 in naredimo vse ustrezne kombinacije, potem postavimo vrednosti koeficientov dveh oglišč na 0 itd. Upoštevati moramo, da vedno ostane vsaj en koeficient večji od 0. S tem dobimo vsa prava lica. Unija vseh teh kombinacij pa je rob.

Simplicialni kompleks K je konča množica simpleksov za katere velja:

- a) $\sigma \in K, \tau \leq \sigma \Rightarrow \tau \in K$;
- b) $\sigma, \sigma' \in K \Rightarrow \sigma \cap \sigma' \leq \sigma, \sigma'$.

Vzemimo, da naš kompleks vsebuje več simpleksov različnih dimenzij. Dimenzija simplicialnega kompleksa K je enaka najvišji dimenziji simpleksa v njem.

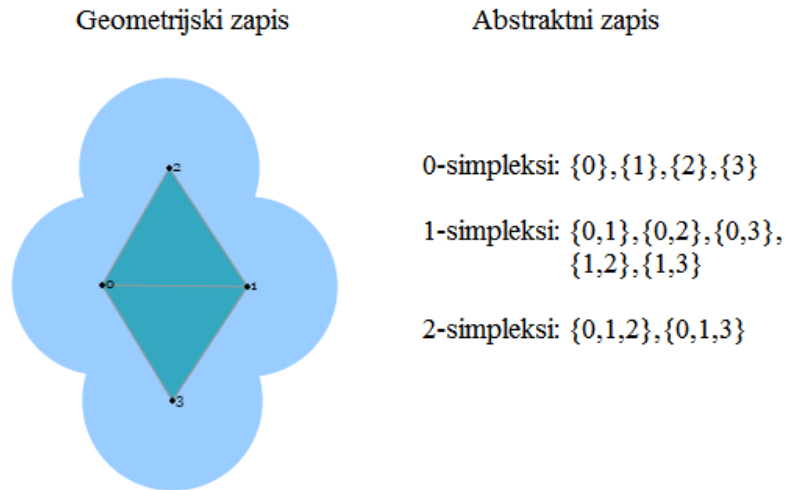
Če se osredotočimo samo na oglišča in na povezave med simpleksi v simplicialnem kompleksu in zanemarimo informacije o tem kje in kako so simpleksi predstavljeni v nekem evklidskem prostoru, govorimo o abstraktnem simplicialnem kompleksu. Podobno je pri senzorskem omrežju, kjer imamo podatke samo o tem, katere senzorje imamo in kateri se med seboj zaznajo. Abstraktni simplicialni kompleks je zato lažje zgraditi in z njimi računati.

Abstraktni simplicialni kompleks A je družina podmnožic neke univerzalne končne množice S , za katere velja: če je $\alpha \in A$ in $\beta \subseteq \alpha$ je tudi $\beta \in A$.

Množice v A predstavljajo abstraktne simplekse. Dimenzija abstraktnega simpleksa je $\dim \alpha = |\alpha| - 1$, kjer je $|\alpha|$ moč množice α . Najvišja dimenzija simpleksa v nekemu kompleksu je hkrati tudi dimenzija kompleksa. Lice abstraktnega simpleksa α je neprazna podmnožica $\beta \subseteq \alpha$.

Danemu geometričnemu simplicialnemu kompleksu K lahko priredimo abstraktni simplicialni kompleks A tako, da odvezamemo koordinate oglišč in obdržimo samo podatke o tem, katere

množice oglišč tvorijo simplekse. Kompleks A imenujemo tudi točkovna shema K . Obratno imenujemo K geometrijska realizacija A . Za primer lahko vzamemo simplicialni kompleks K kot prikazuje Slika 4.



Slika 4: Geometrijski in abstraktni zapis kompleksa

Točkovna shema kompleksa K je zbirka tistih podmnožic množice $\text{Vert } A = \{a_0, \dots, a_n\}$, ki raztezajo simplekse v K , torej

$\{\{a_0, a_1, a_2\}, \{a_0, a_1, a_3\}, \{a_0, a_1\}, \{a_0, a_2\}, \{a_0, a_3\}, \{a_1, a_2\}, \{a_1, a_3\}, \{a_0\}, \{a_1\}, \{a_2\}, \{a_3\}\}$.

Podmnožice $\{a_1, a_2, a_3\}, \{a_0, a_2, a_3\}, \{a_2, a_3\}$ pa ne raztezajo kompleksa K .

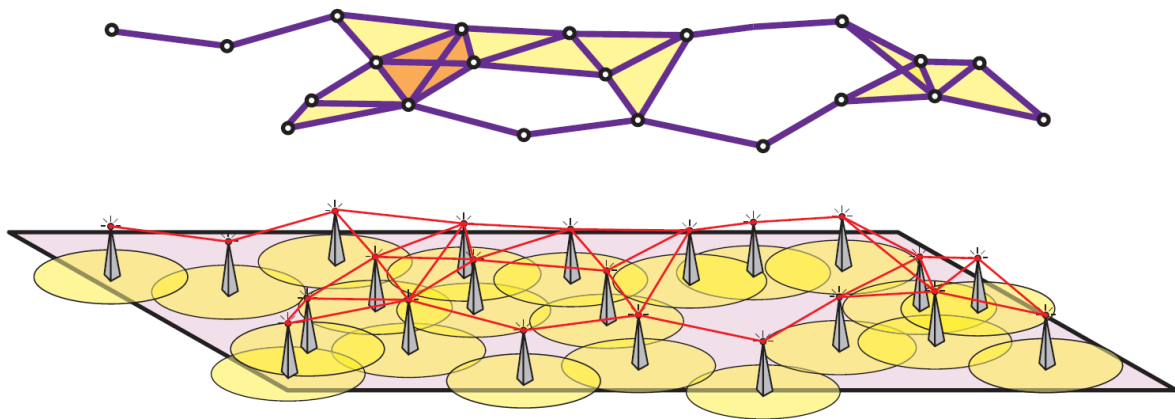
Izrek o geometrijski realizaciji [2, str. 64]: Vsak abstraktni simplicialni kompleks dimenzije d ima geometrijsko realizacijo v prostoru \mathbb{R}^{2d+1} .

2.3 Vietoris-Ripsov kompleks

Vietoris-Ripsov kompleks ali kar Ripsov kompleks je matematični model s katerim iz dane množice točk v prostoru rekonstruiramo obliko prostora, iz katerega so točke zajete [12]. Predstavljajmo si mrežo senzorjev in njegovih povezav, ki »lebdijo« nad realnim prostorom, kot prikazuje Slika 5. Vsaka točka, ki predstavlja položaj enega senzorja ima okolico s polmerom $\epsilon > 0$, ki je za vse točke enak in predstavlja domet posameznega senzorja. Ripsov kompleks je abstraktni simplicialni kompleks \mathcal{R}_ϵ , katerega oglišča so posamezne točke. Simpleksi pa so

množice točk, ki so paroma druga drugi v okolici, torej imajo paroma manjšo razdaljo kot ϵ . Z drugimi besedami, to so točke, ki predstavljajo medsebojno povezane senzorje.

V diplomski nalogi bodo oglišča Ripsovega kompleksa predstavljala senzorje v senzorskem omrežju, polmer okolic bo doseg oddajanja r_b posameznih senzorjev. Z daljicami med točkami označimo, kateri senzorji se med seboj zaznajo. Naš cilj bo zgraditi Ripsov kompleks na podlagi informacij o povezavah med senzorji in izračunati tiste njegove topološke lastnosti, ki so pomembne za lastnosti senzorskega omrežja. Naslednja lema pove, da senzorji v točkah, ki so oglišča simpleksa v Ripsovem kompleksu, v celoti pokrivajo območje, ki ga omejujejo.



Slika 5: Predstavitev Ripsovega kompleksa nad Evklidsko ravnino [8]

Lema: Konveksna ogrinjača množice točk, ki tvorijo simpleks v \mathcal{R} , leži znotraj unije pokritega območja teh točk.

Za dokaz leme bomo potrebovali dobro znani Hellyjev izrek [5]. Dokaz Hellyjevega izreka najdemo na primer v [2, str. 69].

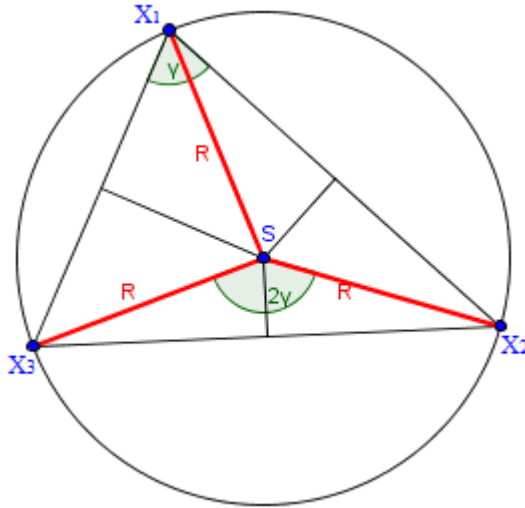
Hellyjev izrek: Naj bo F končni nabor zaprtih konveksnih množic v \mathbb{R}^d . Če ima vsaka množica velikosti $d + 1$ znotraj F neprazen skupen presek, pomeni da imajo tudi vse množice v F neprazen skupen presek.

Zaradi Hellyjevega izreka moramo dokazati le, da je območje, ki ga raztezajo točke simpleksov dimenzije 1 in 2 Ripsovega kompleksa v realnem prostoru, res pokrito.

Predstavljajmo si trojico točk $\{x_1, x_2, x_3\}$, ki označujejo trikotnik s stranicami manjšimi kot r_b (Slika 6). Katerakoli množica diskov, ki se stikajo v skupni točki, vedno vsebuje konveksno lupino teh točk. Torej moramo dokazati, da se trije diski pokritosti s središči v

točkah $\{x_1, x_2, x_3\}$, stikajo v skupni točki. Če je trikotnik topokotni (ali pravokotni), potem je stičišče vseh treh diskov na sredini najdaljše stranice trikotnika, zato $r_c > r_b/2$ zadostuje. Če pa je trikotnik ostrokotni, potem je recimo največji kot γ pri točki x_1 , $\frac{\pi}{3} \leq \gamma \leq \frac{\pi}{2}$ in zato bo $\sin(\gamma) \geq \frac{\sqrt{3}}{2}$. Kot prikazuje Slika 6, lahko radij očrtanega kroga R trikotnika zapišemo kot

$$R = |x_2 - x_3|/2 \sin(\gamma),$$



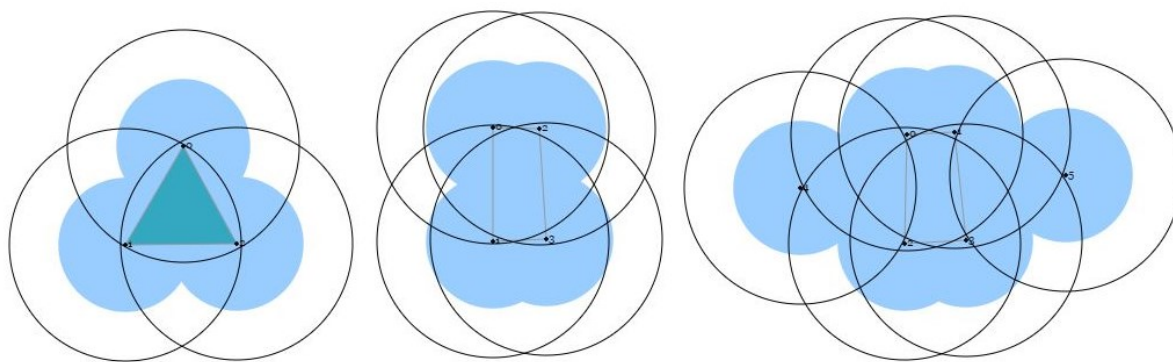
Slika 6: Ostrokotni trikotnik

iz česar lahko izpeljemo $R \leq \frac{r_b}{\sqrt{3}} \leq r_c$.

2.3.1 Opomba

Vsi trije diski v zgornjem dokazu imajo torej skupno točko zaradi predpostavke v diplomski nalogi, da je $r_c \geq r_b/\sqrt{3}$. Razmerje $r_c \geq r_b/\sqrt{3}$ je optimalno, saj velja v primeru enakostraničnega trikotnika z dolžino stranic r_b , enakost $R = r_b/\sqrt{3}$

Celoten Ripsov kompleks na množici točk pa ne zajame vedno topologije unije diskov pokritosti z radijem r_c s središči v teh točkah, kot prikazuje Slika 7. Na levi strani slike je prikazan primer simpleksa dimenzije 2, kjer Ripsov kompleks leži v območju pokritosti. Na sredini in na desni pa sta prikazana primera, kjer območje sicer je pokrito, Ripsov kompleks pa tega ne odraža. V srednjem primeru na sliki je Ripsov kompleks prazen kvadrat, torej območje z luknjo, v primeru na desni pa je na vsaki strani kvadrata dodano še oglišče. Vidimo, da je območje pokrito, vendar Ripsov kompleks ne poveže oglišči s kvadratom. S pomočjo Ripsovega kompleksa torej dobimo potreben, ne pa tudi zadosten pogoj za pokritost območja.



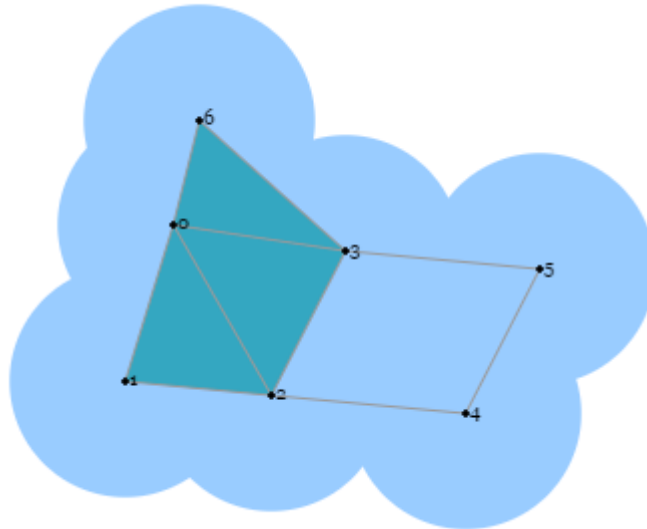
Slika 7: Zajem pokritosti Ripsovega kompleksa

2.4 Homologija

Homologija je način kako matematično opišemo »naluknjanost« prostora. V diplomski nalogi se ukvarjamo s problemom pokritosti omrežja, za katerega želimo ugotoviti ali je območje, ki ga razteza, v celoti pokrito. To območje bomo imenovali domena D . Z drugimi besedami, iščemo »luknje« v omrežju. To so območja, ki jih senzorji ne pokrivajo. Z izgradnjo Ripsovega kompleksa nad tem omrežjem prostor poenostavimo tako, da odmislimo koordinate in se osredotočimo le na povezave med oglišči. Naše omrežje je predstavljeno kot simplicialni kompleks. S pomočjo homologije tega kompleksa pa bomo ugotovili, če v domeni D obstajajo luknje. Konstrukcijo homoloških grup povzemamo po [2, poglavje 4].

2.4.1 Grupe verig

Naj bo K simplicialni kompleks kot na Slika 8. Formalno vsoto simpleksov dimenzije p v K imenujemo p -veriga. Za dimenzijo 2 tako dobimo 2-verige, ki so vsote 2-simpleksov v kompleksu K , na primer $\{0,1,2\} + \{0,2,3\}$. Če označimo z σ_i p -simplekse, lahko p -verigo zapišemo kot $c = \sum a_i \sigma_i$, kjer so a_i koeficienti. Zgornjo 2-verigo splošno zapišemo $1 * \{0,1,2\} + 0 * \{0,3,6\} + 1 * \{0,2,3\}$. Koeficienti a_i , ki jih bomo uporabljali imajo vrednosti 0 ali 1 in določajo kateri simpleksi so v verigi. Za to diplomsko nalogo nam bo zadostovala uporaba koeficientov modula 2.

Slika 8: Kompleks K z 2-verigami

P -verige lahko seštevamo tako, da seštejemo koeficiente pri posameznih simpleksih. Zelo podobno kot pri seštevanju polinomov, kjer seštevamo koeficiente členov z enakimi potencami. Torej, če imamo dve verigi $c = \sum a_i \sigma_i$ in $c' = \sum b_i \sigma_i$, potem je $c + c' = \sum (a_i + b_i) \sigma_i$. Zgoraj smo omenili, da bomo uporabljali modul 2 koeficiente, zato moramo biti pri seštevanju teh pozorni, da velja $1 + 1 = 0$. Zaradi teh koeficientov kot rezultat seštevanja dobimo unijo vseh simpleksov, v katerih se verigi razlikujeta.

Grupo p -verig tvorimo iz p -verig skupaj z operacijo seštevanja in zapišemo kot $(C_p, +)$ ali pa $C_p = C_p(K)$. Grupo p -verig dobimo za vsako dimenzijo p . Grupe verig dimenzij večje kot je dimenzija kompleksa K in manjše kot 0 so trivialne (vsebujejo samo element 0). Zanimajo nas predvsem grupe verig dimenzij 0, 1 in 2, ki bodo dovolj za izračun pokritosti domene D .

Med grupami verig obstaja povezava, ki jo dobimo s preslikavo simpleksov v nižjo dimenzijo. To preslikavo imenujemo robni homomorfizem ∂_p ali na kratko rob in zapišemo kot $\partial_p: C_p \rightarrow C_{p-1}$. Rob verige je seštevek robov njenih simpleksov, $\partial_p c = \sum a_i \partial_p \sigma_i$. Rob simpleksa pa je seštevek vseh njegovih lic dimenzije $p - 1$. Obstaja preprost način za računanje roba simpleksa: rob p -simpleksa $\sigma = \{u_0, u_1, \dots, u_p\}$ je unija vseh podmnožic z $p - 1$ elementi. Drugače povedano, rob je vsota vseh $(p - 1)$ -simpleksov, ki jih raztezajo množice oglišč danega p -simpleksa. Te dobimo tako, da izpustimo po en element v množici oglišč p -simpleksa. Matematično to zapišemo

$$\partial_p \sigma = \sum_{j=0}^p \{u_0, \dots, \hat{u}_j, \dots, u_p\},$$

kjer strešica označuje izpuščen element.

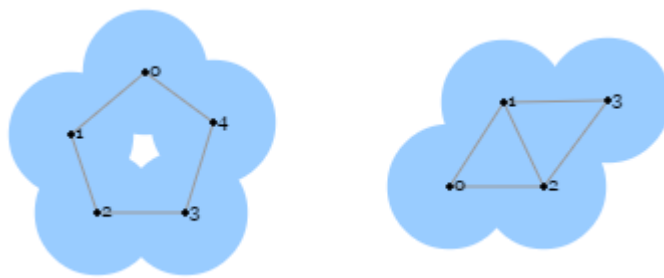
Zaporedju verižnih grup, ki jih povezujejo robni homomorfizmi, pravimo verižni kompleks:

$$\dots \xrightarrow{\partial_{p+2}} C_{p+1} \xrightarrow{\partial_{p+1}} C_p \xrightarrow{\partial_p} C_{p-1} \xrightarrow{\partial_{p-1}} \dots$$

Običajno indeksa robnega homomorfizma ne pišemo, saj se ga da razbrati iz dimenzije verig.

2.4.2 Cikli in robovi

Povedali smo kaj so p -verige, sedaj pa bomo pogledali še dva posebna tipa verig, ki sta pomembna pri računanju homoloških grup. Rob p -verige je $(p - 1)$ -veriga, ki je lahko enaka 0. Če je rob verige enak 0, torej je veriga v jedru robnega homomorfizma, potem tako p -verigo imenujemo p -cikel in jo označimo kot z_p . Vsota dveh p -ciklov je spet p -cikel, zato vsi p -cikli sestavljajo grupo p -ciklov, ki jo označujemo kot $Z_p(K)$ ali krajše Z_p . Vzemimo, da je c_p veriga v kompleksu K . Če velja $\partial c_p = 0$, potem je $c_p \in Z_p$, torej je grupa ciklov ravno jedro robnega homomorfizma $Z_p = \ker \partial_p$. Da si grafično lažje predstavljamo kaj je p -cikel, bomo za primer vzeli kompleksa na Slika 9.



Slika 9: Primer kompleksa s ciklom

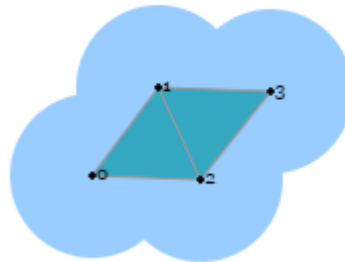
Poglejmo si 1-verigo $\{0,1\} + \{0,4\} + \{1,2\} + \{2,3\} + \{3,4\}$ na levi strani slike. Zaradi enostavnosti kompleksa takoj opazimo, da veriga tvori sklenjeno krivuljo in sklepamo, da spada v grupo 1-ciklov. Tudi vsako oglišče posebej predstavlja 0-cikel in spada v grupo 0-ciklov. Pogosto pa kompleksi niso tako enostavni. Takrat je potrebno določiti p -cikle računsko.

Poglejmo 1-verigo $\{0,1\} + \{0,2\} + \{1,2\} + \{1,3\} + \{2,3\}$ na desni strani slike. Na prvi pogled bi lahko rekli, da tudi ta veriga spada v grupo 1-ciklov, vendar temu ni tako zaradi dodatne povezave med oglišči 1 in 2. Pa si pogledjmo to računsko:

$$\begin{aligned}
 \partial(\{0,1\} + \{0,2\} + \{1,2\} + \{1,3\} + \{2,3\}) &= \\
 &= \partial\{0,1\} + \partial\{0,2\} + \partial\{1,2\} + \partial\{1,3\} + \partial\{2,3\} = \\
 &= 0 + 1 + 0 + 2 + 1 + 2 + 1 + 3 + 2 + 3 = \\
 &= 1 + 2
 \end{aligned}$$

Dobimo, da je rob verige neničelna veriga dimenzije 0. To pa ne pomeni, da ne moremo najti 1-ciklov v desnem kompleksu. 1-verige $\{0,1\} + \{0,2\} + \{1,2\}$, $\{1,2\} + \{1,3\} + \{2,3\}$ in $\{0,1\} + \{0,2\} + \{1,3\} + \{2,3\}$ spadajo v grupo 1-ciklov. Grupa p -ciklov je vedno podmnožica grupe p -verig $Z_p \subseteq C_p$ (Slika 3).

Če je p -veriga rob višje dimenzionalne $(p + 1)$ -verige, p -verigo imenujemo p -rob in označimo kot b_p . Podobno kot p -cikli, lahko tudi p -robovi sestavljajo grupo. Grupo p -robov kompleksa K zapišemo kot $B_p(K)$ ali krajše B_p in je enaka sliki $(p + 1)$ -tega robnega homomorfizma $B_p = \text{im } \partial_{p+1}$. Vzemimo, da je c_{p+1} veriga v kompleksu K . Če velja $\partial c_{p+1} = c_p$, potem je $c_p \in B_p$. Poglejmo si kompleks na Slika 10.



Slika 10: Primer kompleksa z 1-robom

Kompleks je podoben tistemu na Slika 6, le da smo dodali še simpleksa dimenzije 2. Izračunajmo rob 2-verige c_2 , ki vsebuje oba simpleksa dimenzije 2, $c_2 = \{0,1,2\} + \{1,2,3\}$. Rob 2-verige je enak:

$$\begin{aligned}
 \partial c_2 &= \partial\{0,1,2\} + \partial\{1,2,3\} = \\
 &= \{0,1\} + \{0,2\} + \{1,2\} + \{1,2\} + \{1,3\} + \{2,3\} = \\
 &= \{0,1\} + \{0,2\} + \{1,3\} + \{2,3\} = b_1.
 \end{aligned}$$

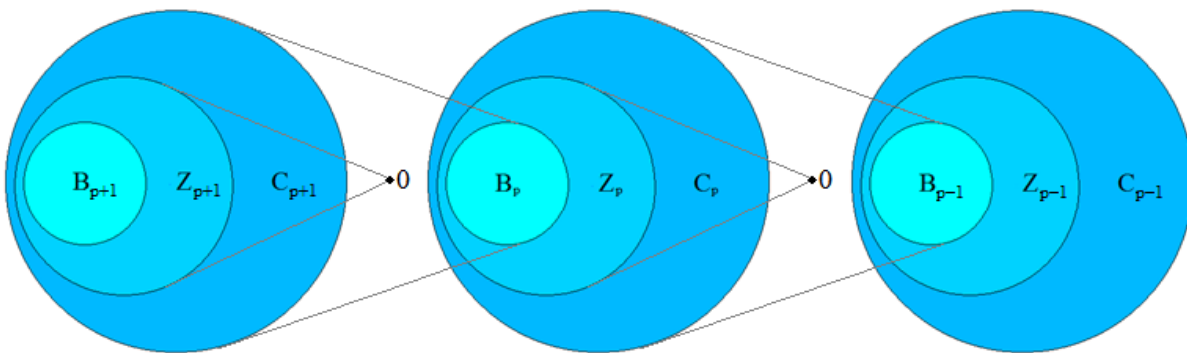
Dobili smo verigo dimenzije 1, ki spada v grupo 1-robov, $b_1 \in B_1$. Grafično si lahko rob verige predstavljamo kot unijo za 1 dimenzijo nižjih simpleksov, ki obdajajo to verigo. V našem primeru so to vse stranice, ki obdajajo oba trikotnika, brez skupne stranice.

Grupa p -robov je podgrupa v grupi p -verig $B_p \subseteq C_p$. Hkrati pa je podgrupa tudi grupe p -ciklov, zaradi naslednjega izreka:

Osnovni izrek homologije [2, str. 95]:

$$\partial_p \partial_{p+1} d = 0 \text{ za vsak } p \in \mathbb{N} \text{ in vsako } (p+1) - \text{verigo } d.$$

Izrek nam pove, da je rob roba vedno 0, zato je $B_p \subseteq Z_p \subseteq C_p$ ali pa, za cel verižni kompleks, $B_{p+1} \subseteq Z_{p+1} \subseteq C_{p+1} \xrightarrow{\partial_{p+1}} B_p \subseteq Z_p \subseteq C_p \xrightarrow{\partial_p} B_{p-1} \subseteq Z_{p-1} \subseteq C_{p-1}$, (Slika 11).



Slika 11: Homomorfizem in grupe

2.4.3 Homološke grupe

Homološke grupe nam povedo, koliko lukenj različnih dimenzij je v danem objektu. Ker pa je luknja tam, kjer ničesar ni, ni bilo očitno kako jo formalno definirati in ločiti med različnimi vrstami. Homologija je matematična metoda za definiranje in kategoriziranje lukenj po oblikah.

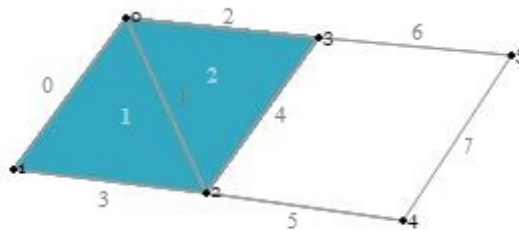
Definicija: p -ta homološka grupa je kvocientna grupa grupe p -ciklov in grupe p -robov: $H_p = Z_p/B_p$. Rang grupe H_p imenujemo p -to Betti število, $\beta_p = \text{rang } H_p$.

Homološka grupa H_p vsebuje vse p -cikle, vendar ne razločuje med cikloma, ki se razlikujeta za rob neke $(p + 1)$ -verige. Za to diplomsko nalogo bo dovolj, če poznamo samo njen rang, torej Bettijevo število. Rang grupe je maksimalno število linearno neodvisnih generatorjev grupe in ga lahko izračunamo kot razliko med rangoma grupe ciklov Z_p in grupe robov B_p , torej $\beta_p = \text{rang } H_p = \text{rang } Z_p - \text{rang } B_p$. Za dimenzije med 0 in dimenzijo kompleksa nam to število pove, koliko različnih lukenj te dimenzije vsebuje simplicialni kompleks. Bettijevo število dimenzije 0 pa ima drugačen pomen. Pove nam število povezanih komponent simplicialnega kompleksa.

2.5 Postopek računanja

Za primer računanja bomo vzeli simplicialni kompleks K , predstavljen na

Slika 12. Senzorji so označeni s črno barvo, povezave s sivo in pokrita površina z modro.



Slika 12: Simplicialni kompleks K

Računali bomo s pomočjo matrik. Iz simplicialnega kompleksa lahko preberemo incidenčne matrike oz. matrike sosednosti, ki opisujejo robne homomorfizme. Povedo nam, kakšen je odnos med dvema razredoma objektov. V našem primeru bodo razredi simpleksi različnih dimenzij. Če za prvi razred vzamemo simpleks dimenzije 0 ter ga označimo z V (Vertex), in za drugi razred vzamemo simpleks dimenzije 1 ter ga označimo z E (Edge), potem ima matrika eno vrstico za vsak element razreda V in en stolpec za vsak element razreda E . V matriki je element v vrstici V in stolpcu E enaki 1, če je oglišče V lice stranice E .

									T0 T1	
E0 E1 E2 E3 E4 E5 E6 E7									E0	E1
V0	1	1	1	0	0	0	0	0	1	0
V1	1	0	0	1	0	0	0	0	1	1
V2	0	1	0	1	1	1	0	0	0	1
V3	0	0	1	0	1	0	1	0	1	0
V4	0	0	0	0	0	1	0	1	0	0
V5	0	0	0	0	0	0	1	1	0	0
									E6	E7
									0	0

Slika 13: Incidenčni matriki simpleksov dimenzije 0 in 1 (na levi) ter dimenzij 1 in 2 (na desni). Z V so označeni 0-simpleksi (točke), z E 1-simpleksi (daljice), s T pa 2-simpleksi (površina trikotnikov).

Za izračun pokritosti bomo potrebovali 2 matriki. Prva bo incidenčna matrika simpleksov dimenzije 0 in 1, druga pa simpleksov dimenzije 1 in 2 (Slika 13). Matrike nam same po sebi ne pokažejo potrebnih informacij, zato jih je treba preoblikovati. Vrstice matrike ustrezajo simpleksom dimenzije p , ki tvorijo bazo grupe C_{p-1} , stolpci pa bazo grupe C_p . Z uporabo stolpčnih in vrstičnih operacij lahko matriko pretvorimo v Smith normalno obliko in hkrati ohranimo rang matrike nespremenjen.

Smithova normalna oblika ima na diagonali v prvih nekaj vrsticah enice, ki se začnejo v levem zgornjem kotu, ostala polja matrike pa so 0 (Slika 14). Iz tako reducirane oblike, lahko preberemo rang grupe ciklov $Z_p = \ker \partial_p$ in grupe robov $B_p = \text{im } \partial_p$. Število stolpcev n_p je enako številu simpleksov dimenzije p , kar je enako rang grupe C_p . Število z_p je enako številu ničelnih stolpcev C_p , kar je rang grupe p -ciklov (Slika 14). Število neničelnih vrstic predstavlja $b_{p-1} = \text{rang}(B_{p-1})$, rang grupe $(p-1)$ -robov. Iz linearne algebre vemo, da velja $n_p = b_{p-1} + z_p$, zato je v matriki b_{p-1} stolpcev, ki imajo vrednost 1 po diagonali in z_p ničelnih stolpcev. V našem primeru predstavlja prvih 5 stolpcev p -verigo, katere neničelni robovi generirajo grupo $(p-1)$ -robov. Ostali stolpci pa predstavljajo p -cikle, ki generirajo Z_p .

		rang C_p					rang Z_p		
		(E0)	(E0+E1)	(E1+E2)	(E1+E2+E5)	(E1+E2+E5+E6)	(E0+E1+E3)	(E1+E2+E4)	(E1+E2+E5+E6+E7)
rang C_{p-1}	(V1+V0)	1	0	0	0	0	0	0	0
	(V2+V1)	0	1	0	0	0	0	0	0
	(V3+V2)	0	0	1	0	0	0	0	0
	(V4+V3)	0	0	0	1	0	0	0	0
	(V5+V4)	0	0	0	0	1	0	0	0
	(V5)	0	0	0	0	0	0	0	0

Slika 14: Reducirana incidenčna matrika dimenzije 0 in 1. Število stolpcev je enako številu simpleksov dimenzije p , število vrstic pa je enako številu simpleksov dimenzije $(p - 1)$.

Rang grupe ciklov je enak številu ničelnih stolpcev. Rang grupe robov je enak številu neničelnih vrstic.

		rang C_{p+1}	
		rang B_{p+1}	
		(T0)	(T1)
rang C_p	(E3+E1+E0)	1	0
	(E4+E2+E1)	0	1
	(E2)	0	0
	(E3)	0	0
	(E4)	0	0
	(E5)	0	0
	(E6)	0	0
	(E7)	0	0

Slika 15: Reducirana incidenčna matrika dimenzije 1 in 2. Število stolpcev je enako številu simpleksov dimenzije $(p + 1)$, število vrstic pa je enako številu simpleksov dimenzije p .

Število neničelnih vrstic je enako rangi grupe robov.

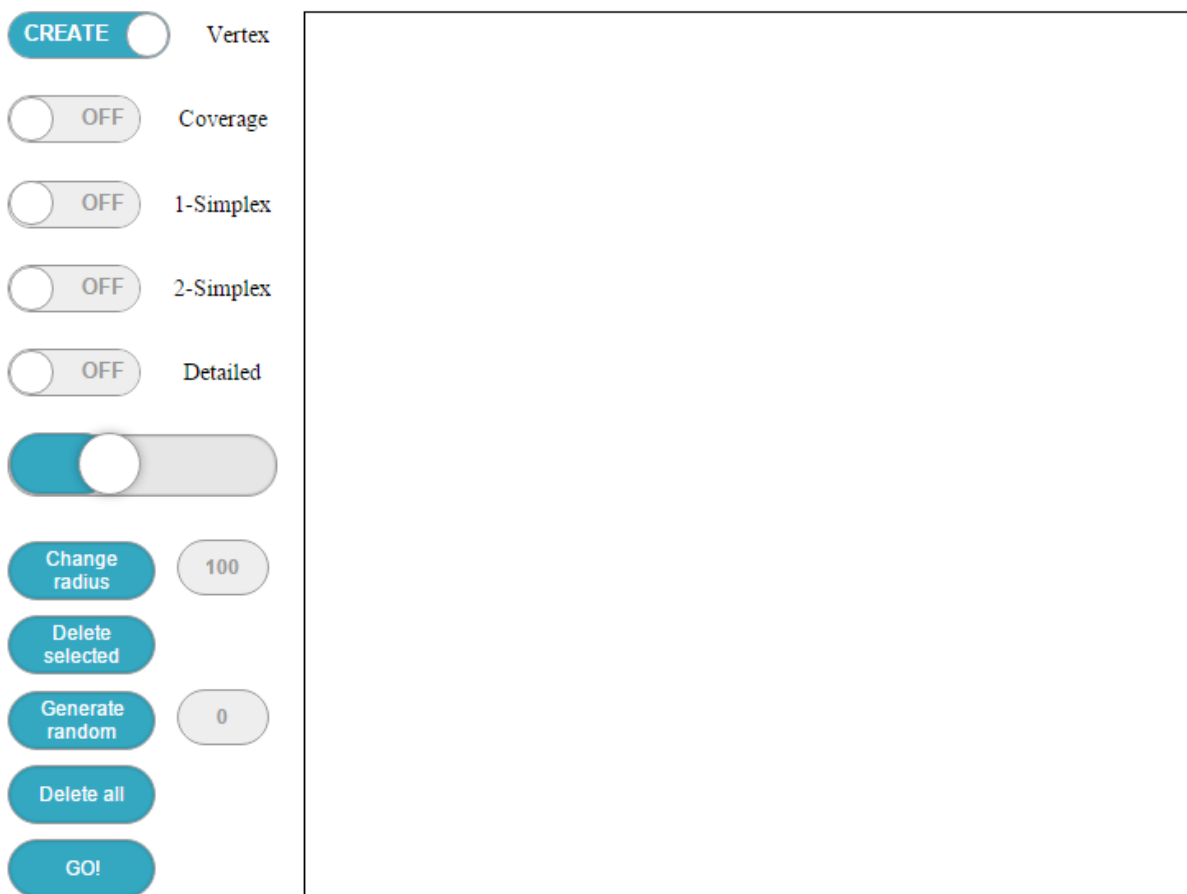
Ko so vse matrike v Smith-ovi normalni obliki, lahko izračunamo Bettijeva števila kot razliko med rangi $\beta_p = \text{rang}(Z_p) - \text{rang}(B_p)$, $p \geq 0$. Bolj natančno nas zanimata le dve Bettijevi števili $\beta_0 = \text{rang}(H_0)$ in $\beta_1 = \text{rang}(H_1)$. Ničto Bettijevo število nam pove, koliko povezanih komponent ima simplicialni kompleks, prvo pa koliko različnih 1-dimenzionalnih lukenj je v njem.

Torej je $\beta_1 = \text{rang}(H_1) = \text{rang}(Z_1) - \text{rang}(B_1)$. Povedali smo, da je $\text{rang}(Z_1)$ enak številu ničelnih stolpcev v reducirani incidenčni matriki dimenzij 0 in 1, kar je v našem primeru enako $z_1 = 3$ (Slika 14). Rang(B_1) pa je enak številu neničelnih vrstic v reducirani incidenčni matriki dimenzij 1 in 2, kar je v našem primeru enako $b_1 = 2$ (Slika 15). Izračunamo $\beta_1 = \text{rang}(H_1) = 3 - 2 = 1$, kar nam pove, da imamo v našem sistemu eno 1-dimenzionalno luknjo. Torej domena omrežja, ki ga opisuje naš simplicialni kompleks, ni pokrita.

Poglavje 3 Simulator

3.1 Opis

Program smo napisali v jeziku Java Script. To je objektno skriptni programski jezik, ki ga je razvil Netscape. Njegova posebnost je poenostavitev izdelave interaktivnih spletnih strani. HTML kodo poživi z dinamičnim izvajanjem in interaktivnostjo. Pomembno je tudi, da ga podpirajo vsi novejši spletni brskalniki, kar zagotovi dostopnost.



Slika 16: Izgled aplikacije

Aplikacija služi kot simulator izračuna pokritost senzorskega omrežja s topološkimi metodami. Posebnost teh pa je, da se pokritost računa samo na podlagi povezav med senzorji, zato dejanskih koordinat oz. razdalj med njimi ne potrebujemo. Aplikacija je enostavna in preprosta za uporabo (Slika 16). Na levi imamo meni gumbov, ki služi za upravljanje prikaza podatkov in spreminjanju določenih vrednosti, na desni pa imamo ponjavo, ki služi za grafično prikazovanje. Pod vsem tem pa se izpisujejo še matrike, ki so uporabljene pri računanju pokritosti.

CREATE/SELECT in DELTE, kako deluje? Na ponjavi z navadnim klikom miške označimo mesto senzorjev. Ti se izrišejo na mestu klika kot točke, ki so oštevilčene po zaporedju klika. Prvi senzor bo označen z 0, drugi z 1 itd. V primeru, da želimo mesto senzorja premakniti, ga lahko izberemo in izbrišemo. To naredimo tako, da v meniju kliknemo na preklopni gumb »CREATE/SELECT«, da se nam pokaže opcija »SELECT«. S klikom na ustrezno točko željeni senzor izberemo. Izbrani senzor bo obrobjen z rdečo in ga izbrišemo s pritiskom na gumb »Delete selected«. Ostali senzorji pa bodo ponovno oštevilčeni, tokrat brez izbrisanega senzorja. Izbrišemo lahko tudi vse senzorje hkrati, s pritiskom na gumb »Delete all«.

Generiramo lahko tudi izbrano število senzorjev, ki bodo postavljeni na naključno mesto. V polje zraven gumba »Generate random« vpišemo število senzorjev, ki jih želimo generirati. S pritiskom na gumb »Generate random«, se bodo ti naključno pojavili na ponjavi. Na tem mestu bi radi opozorili, da program ni optimiziran. Za računanje pokritosti ne priporočamo generiranje več kot 100 senzorjev.

Vsak senzor lahko zazna in se poveže z drugimi senzorji, če so v njegovem dosegu in obratno. Radij dosega lahko spreminjamo z gumbom »Change radius« ali pa z drsnikom nad njem. V polje, desno od gumba »Change radius«, vnesemo željeno velikost radija in vnos potrdimo s klikom na gumb. Pri uporabi drsnika, se velikost radija samodejno spreminja glede na njegovo pozicijo in se prikazuje v istem polju, ki smo ga zgoraj uporabili za direkten vnos.

Zaradi preglednosti, je mogoče grafične prikaze posebej vklopiti in izklopiti v meniju (Slika 17):

- »Coverage«

Vključimo s pritiskom na preklopni gumb »ON/OFF«, poleg napisa »Coverage« tako, da se nam pokaže opcija »ON«. Ta nam prikazuje doseg senzorja, to je območje znotraj črno obarvane krožnice, ki ima za središče senzor in njegovo pokritost, ki je prikazana kot območje svetlo modre barve okoli senzorja.

- »1-Simplex«

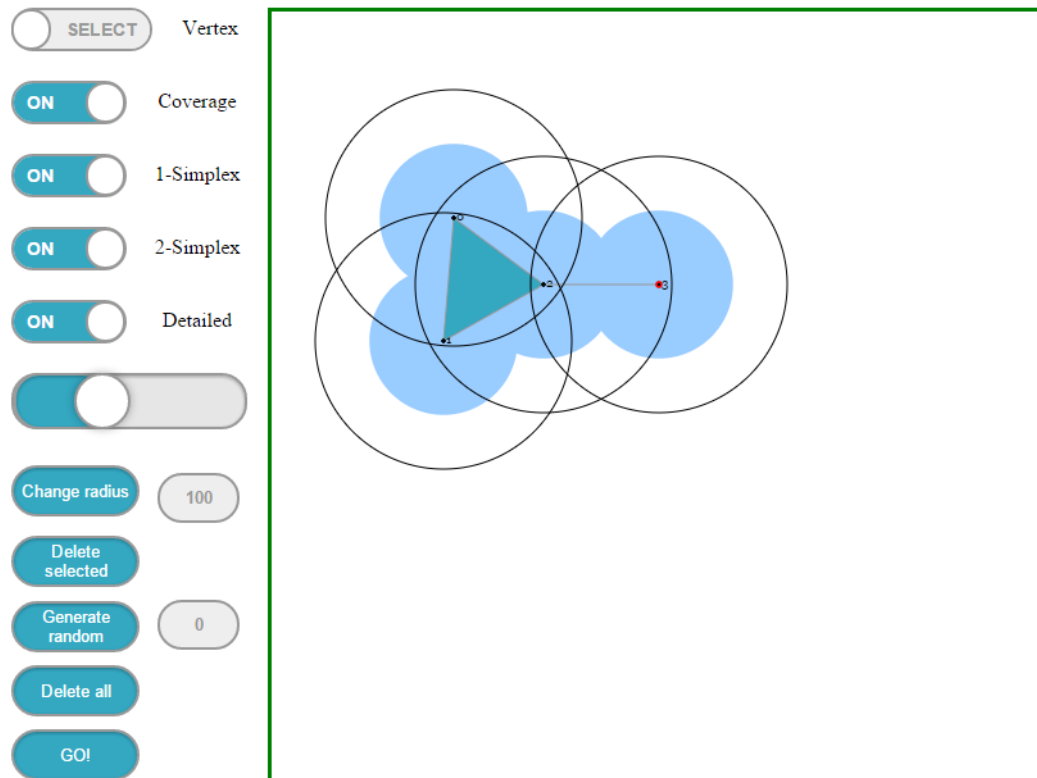
Vključimo s pritiskom na preklopni gumb »ON/OFF« poleg napisa »1-Simplex« tako, da se nam pokaže opcija »ON«. Ta nam prikazuje vse simplekse dimenzije 1, kar so v bistvu daljice med točkami. Aplikacija poveže vse senzorje, ki se med seboj zaznajo s sivo daljico, ki prikazuje povezavo.

- »2-Simplex«

Vključimo s pritiskom na preklopni gumb »ON/OFF« poleg napisa »2-Simplex« tako, da se nam pokaže opcija »ON«. Izrišejo se nam simpleksi dimenzije 2, to so polni trikotniki modro-zelene barve, ki predstavljajo pokrito območje.

Sedaj, ko je kompleks narisan, pritisnemo gumb »GO!«. Aplikacija bo ustvarila matrike in izračunala pokritost. Okvir ponjave se bo obarval glede na izračunane rezultate. Rumen okvir nam pove, da vsi elementi niso med seboj povezani. Pokritosti v temu primeru ni smiselno računati. Zelen okvir pomeni, da je kompleks pokrit, rdeč pa oznani vsaj eno luknjo v kompleksu. Na dnu se izpišeta tudi Bettijevi števili β_1 in β_0 , ter rang grupe ciklov in grupe robov za dimenziji 0 in 1. Za prikaz incidenčnih matrik uporabimo preklopna gumba »ON/OFF« za grafični prikaz »1-Simplex« in »2-Simplex«, ki skupaj s preklopnim gumbom »ON/OFF« zraven napisa »Detailed« določajo, katere informacije o kompleksnem sistemu naj se izpišejo (Slika 17). Torej če želimo:

- izpisati incidenčno matriko dimenzij 0 in 1, bomo preklopni gumb pri »1-Simplex« nastavili na »ON«.
- izpisati incidenčno matriko dimenzij 1 in 2, bomo nastavili preklopni gumb pri »2-Simplex« na »ON«.
- izpisati normalno obliko zgoraj navedenih matrik, pa nastavimo preklopni gumb pri napisu »Detailed« na »ON«.



$H(1)=0$
 $Z1=1$ $B1=1$

$H(0)=1$
 $Z0=4$ $B0=3$

$T0$
 $E0$ 1
 $E1$ 1
 $E2$ 1
 $E3$ 0

$(T0)$
 $(E2+E1+E0)$ 1
 $(E1)$ 0
 $(E2)$ 0
 $(E3)$ 0

$E0$ $E1$ $E2$ $E3$
 $V0$ 1 1 0 0
 $V1$ 1 0 1 0
 $V2$ 0 1 1 1
 $V3$ 0 0 0 1

$(E0)$ $(E0+E1)$ $(E3)$ $(E0+E1+E2)$
 $(V1+V0)$ 1 0 0 0
 $(V2+V1)$ 0 1 0 0
 $(V3+V2)$ 0 0 1 0
 $(V3)$ 0 0 0 0

Slika 17: Izgled aplikacije z grafičnim prikazom »Coverage«, »1-Simplex« in »2-Simplex«. Točka 3 je izbrana, spodaj pa so izpisane incidenčne matrike in njihove normalne oblike.

3.2 Grafična predstavitev

Za grafično predstavitev in vnos podatkov smo uporabili HTML element `canvas`. Na njega smo pripeli poslušalec dogodkov, ki se je prožil ob kliku na `canvas`. Tako smo lahko beležili mesto klika. Glede na vrednost preklopnega gumba »CREATE/SELECT« se nato kliče metoda za kreiranje senzorja »createVertex«, ali pa za njegovo izbiro »selectVertex«.

Metoda »createVertex« se kliče ob kliku na ponjavo, če je ima preklopni gumb »CREATE/SELECT« vrednost »CREATE«. Poskrbi, da se na mestu klika ustvari nov senzor in posodobi Ripsov kompleks. Koordinate novo narejenih senzorjev se shranijo v tabelo »vertices«. Nato pa se začne gradnja Vietoris-Rips kompleksa s klicem metode »create1Simplex« in »create2Simplex«. Ti ustvarita dvodimenzionalni tabeli »sim1« in »sim2«, v katerih sta opisani dimenziji 1 in 2 našega kompleksa. Višje dimenzije nas ne zanimajo, zato na temu mestu tudi ustavimo gradnjo Ripsovega kompleksa.

Prva dimenzija obeh tabel predstavlja simplekse, druga pa njihove osnovne elemente, ki jih gradijo. Ker programu direktno ne povemo kateri senzorji se med seboj zaznajo in kateri ne, se za to uporabi njihove koordinate in na podlagi njihove medsebojne razdalje ugotovil povezave. V zanki se za vsak par senzorjev iz tabele »vertices« izračuna njuno razdaljo. Če je razdalja manjša kot radij zaznave senzorja, se v tabelo »sim1« zapiše nova povezava. Šele ko je tabela »sim1« dokončana, se preveri katere povezave lahko povežemo v trikotnike.

Iskanje trikotnikov se torej vrši s pomočjo prejšnje tabele. Tokrat paroma med seboj primerjamo povezave iz tabele »sim1«. Če imata obe povezavi skupni senzor, se pogleda, ali obstaja takšna povezava, ki povezuje druga 2 senzorja. V primeru, da tabela »sim1« vsebuje takšno povezavo, dobimo trikotnik oz. 2-simpleks, ki ga zapišemo v »sim2«. Na koncu se kliče metoda za izrisovanje »reDraw«.

Metoda »reDraw«, skrbi za pravilen izpis in izris vseh elementov. Na podlagi preklopnih gumbov v meniju poskrbi za pravilne klice metod izrisovanja kot so:

- »drawCoverage«
Metoda skrbi za izris krogov, ki predstavljajo zaznavanje in pokritost senzorjev. Vrši se v dveh stopnjah, katere določa podani parameter. Prva stopnja se bo izvršila, če je podan parameter 0. Takrat se sprehodi čez tabelo »vertices« in za vsak senzor izriše njegovo pokritost, kot krog modre barve s središčem na mestu senzorja. Podobno naredi tudi druga stopnja, ki se izvrši, če je podan parameter 1. Za vsak senzor v tabeli izriše očrtan krog, ki predstavlja doseg, v katerem lahko senzor zazna druge senzorje.
- »draw2Simplex«

Metoda skrbi za izris 2-simpleksov oz. trikotnikov. Za vsak trikotnik v tabeli »Sim2«, pridobi koordinate senzorjev iz tabele »vertices«, ki ga sestavljajo. Znotraj teh koordinat nariše trikotnik zeleno-modre barve.

- »draw1Simplex«
Metoda skrbi za izris 1-simpleksov oz. povezav med senzorji. Za vsako povezavo v tabeli »Sim1« pridobi koordinati senzorjev iz tabele »vertices«, ki ju povezuje in nariše daljico med njima.
- »draw0Simplex«
Metoda skrbi za izris 0-simpleksa oz. senzorjev. Na koordinatah vsakega senzorja v tabeli »vertices« naredi piko in napiše številko senzorja. Številka senzorja pa je enaka indeksu s katerim je naslovljen.
- »drawSelected«
Metoda skrbi za izris rdečega kroga okoli izbranega senzorja. Če ta obstaja, prebere njegove koordinate iz tabele »closest«. Koordinate služijo kot središče krožnice, ki jo izriše metoda. Tabela »closest« ustvari metoda »selectVertex« po proženju dogodka klik na ponjavi, če je preklopni gumb »CREATE/SELECT« nastavljen na »SELECT«.

Vrstni red izvršitve teh metod je zelo pomemben, saj določa kateri elementi so postavljeni v ospredje in kateri v ozadje. Aplikacija najprej nariše modre kroge, ki predstavljajo pokritost. Nato sledi izris simpleksov dimenzije 2 in 1. Na koncu pa se izrišejo še krožnice dosega in točke, ki predstavljajo senzorje.

3.3 Funkcionalnost

Na levi strani je poleg ponjave narejen meni z gumbi. Nekateri služijo za grafični prikaz, drugi pa za funkcionalnost aplikacije. Gumbe za grafični prikaz in njihove metode, smo opisali že zgoraj. Sedaj pa se bomo posvetili še funkcijskim gumbom:

- »Chane radius« in drsnik
Eden izmed funkcijskih gumbov je »Change radius«, ki z drsnikom nad njim določa velikost dosega in hkrati tudi pokritosti senzorjev. Velikost dosega je zapisana v spremenljivki »threshold«. Po pritisku na gumb se kliče metoda »changeThreshold«, ki prebere vrednost iz vnosnega polja desno od gumba in jo zapiše v spremenljivko »threshold«. Metoda nato poskrbi, da se tabeli »sim1« in »sim2« ponovno generirata in izrišeta s klicem metod »create1Simplex«, »create2Simplex« in »reDraw«.

Velikost dosega pa lahko spreminjamo tudi z drsnikom nad gumbom »Change radius«, ki služi za vnos vrednosti spremenljivke »threshold« na grafičen način. Na drsnik je pripet poslušalec dogodkov, ki kliče metodo »changeThreshold« ob spremembi njegove vrednosti.

- »Generate random«
Applikacija ima možnost generiranja določenega števila naključnih senzorjev. To so senzorji katerih koordinate so generirane naključno z generatorjem naključnih števil v knjižnici »Math«. S pritiskom na gumb »Generate random« se kliče metoda »generateVertices«. Metoda prebere vrednost vnosnega polja desno od gumba, ki pove koliko senzorjev se bo generiralo. V zanki se za vsak senzor generirata 2 naključni števili med 0 in 600, ki se shranita v tabelo »vertices«. V tabeli ti dve števili predstavljata koordinati x in y novo generiranega senzorja. Na koncu se kličejo še metode »create1Simplex«, »create2Simplex« in »reDraw«.
- »Delete selected«
Izbrano točko lahko s pritiskom na gumb »Delete selected« izbrišemo. Po pritisku na gumb se kliče metoda »deleteVertex«, ki preveri, če imamo izbrano točko. To naredi tako, da pogleda v tabelo »closest«. Če je na indeksu 1 v tabeli pozitivno število pomeni, da število predstavlja indeks senzorja, ki se nahaja v tabeli »vertices«, drugače pa senzor ni izbran. V primeru, da je senzor izbran, ga izbriše iz tabele »vertices«. Na koncu pa kliče metode »create1Simplex«, »create2Simplex« in »reDraw«.
- »Delete all«
Opcija, da izbrišemo vse senzorje naenkrat, se izvrši s pritiskom na gumb »Delete all«. Metoda ponastavi tabelo »vertices« na prazno in kliče metode »create1Simplex«, »create2Simplex« in »GO«.
- »GO!«
Preoblikovanje matrik, računanje homologije in tekstovni izpis se izvede s pritiskom na gumb »GO!«, ki povzroči klic metode »GO«. Ta poskrbi, da se izračuna pokritost in izpis potrebnih informacij. Najprej pogleda, če je računanje sploh možno, tj. da tabeli »sim1« in »sim2« sploh obstajata. V primeru, da je tabela »sim1« večja od 0, se kličejo metode v sledečem zaporedju in vsaka posreduje podatke naslednji:
 1. »getMatrix« dobi v parametru podani velikosti tabele »sim1« kot x , in velikost tabele »vertices« kot y .
Metoda generira matriko samih ničel velikosti $x \times y$. Matrika je predstavljena kot dvorazsežnostna tabela »Matrix«, katere prva dimenzija predstavlja premik

po vrsticah, druga pa po stolpcu. S pomočjo dvojne zanke, pa se sprehodi čez tabelo in v vsako celico zapiše vrednost 0. Na koncu nam vrne tabelo »Matrix«.

2. »MVertexSim1«

Metoda napolni podano matriko z enicami tako, da dobimo incidenčno matriko dimenzij 0 in 1. Z dvojno zanko se sprehodi čez tabelo »sim1«. Za naslavljanje celic tabele »Matrix« v katere se zapiše 1, se za prvo dimenzijo uporabljajo indeksi kot za tabelo »sim1«, za naslavljanje druge dimenzije pa se uporabi vrednosti tabele »sim1«. Na koncu nam vrne izpolnjeno tabelo »Matrix« (Slika 13).

3. »Krn1« in »Krn2«

Metodi preoblikujeta podano matriko v Smith normalno obliko (Slika 14 in Slika 15). Prva se izvede metoda »Krn1«, ki vrši operaciji seštevanja in menjavo med stolpci. Druga pa »Krn2«, ki vrši isti operaciji med vrsticami. Obe metodi se izvajata na podoben način; najprej se ugotovi pivotni stolpec oz. vrstico in se ga premakne na pravo mesto, potem pa ga prištejemo drugim. Za seštevanje skrbi metoda »Add«, ki dobi v parametru podana stolpca. Ta sešteje po modulu 2 in vrne njuno vsoto. V tabelo »P« pa se beležijo narejene operacije in indeksi vrstice ali stolpca nad katerim se operacija vrši. Tako na koncu preoblikovanja vemo, katere vrstice ali stolpci so bili seštet in kam.

Za preoblikovanje v Smith normalno obliko se v zanki sprehodimo čez vse vrstice in stolpce. Metoda »Krn1« začne v zgornjem levem kotu in nadaljuje desno, dokler ne najde 1. Tako določi pivotni stolpec za prvo vrstico. Zamenja ga s stolpcem, ki je na prvem mestu, nato pa ga prišteje vsem ostalim stolpcem, ki imajo v tej isti vrstici 1. Po modulu 2 dobimo na najbolj levem mestu v matriki 1, povsod drugod pa 0. Premakne se v naslednjo vrstico in ponovno najde pivotni stolpec, vendar ta ne sme biti eden izmed že uporabljenih. To naredi za vsako vrstico. Vsaka operacija, ki se izvrši, pa se zabeleži v tabelo »P«. Ta je narejena na začetku metode in hrani indekse stolpcev. Če se stolpca v matriki »Matrix« seštejeta ali zamenjata, potem se to naredi tudi v tabeli »P«.

Metoda »Krn2« deluje na zelo podoben način, le da se sprehodi čez vse stolpce in išče pivotne vrstice; torej dela operacije nad vrsticami. Najbolj opazna razlika med njima pa je, da se operacije beležijo nekoliko drugače. Pri metodi »Krn1« se seštevanje 2 stolpcev zapiše v stolpec kateremu smo prištevali pivotni stolpec. Metoda »Krn2« pa v tabelo »P2« shrani seštevek ravno obratno. Indeks vrstic, ki jih seštejemo, se shranita v pivotno vrstico.

Na koncu nam metodi vrneta podano matriko v Smith normalni obliki.

Ko dobimo reducirano matriko, preštejemo število ničelnih stolpcev, ki shranimo v spremenljivko »Zp« in število neničelnih vrstic, ki shranimo v spremenljivko »B0«. Prva predstavlja rang ciklov dimenzije 1, druga pa rang robne grupe dimenzije 0.

Podobno kot za tabelo »sim1« sedaj naredimo še za tabelo »sim2«. Preverimo, da je dolžina tabele večja od 0 in kličemo metode v istem zaporedju kot zgoraj, le da namesto metode »MVertexSim1« kličemo »MSim1Sim2«.

Metoda »MSim1Sim2«, tako kot »MVertexSim1«, napolni podano matriko z enicami, da dobimo incidenčno matriko, vendar tokrat dimenzij 1 in 2. Ko dobimo reducirano matriko preštejemo število neničelnih vrstic, ki jih shranimo v spremenljivko »B1«. Ta pa predstavlja rang robne grupe dimenzije 2.

Na koncu metoda izračuna še Bettijeva števila, obarva rob ponjave glede na rezultate in kliče metodo »reDraw«.

Poglavje 4 Zaključek

Zaradi razvoja tehnologije se je uporaba brezžičnih senzorskih omrežji razširila. Senzorji so se začeli uporabljati na različnih področjih, kot so zagotavljanje varnosti, nadzora, zdravja, spremljanje okolja itd. Sama kvaliteta zaznavanja senzorskih omrežji pa je odvisna od tega, kako dobro je željena domena pokrita. S pomočjo informacije o povezavah med senzorji, nad omrežjem zgradimo komunikacijski graf in Ripsov kompleks. S topologijo nad kompleksom pa izračunamo število lukenj v omrežju. Način računanja pokritosti, ki smo ga predstavili je centraliziran in pri večjemu številu senzorjev zahteva zmogljivo centralno procesno enoto. V večjih omrežjih se zato raje uporablja pristop porazdeljenega računanja, ki pa ima tudi svoje pomanjkljivosti [11].

V diplomski nalogi je predstavljen preprost simulator, ki s pomočjo računske topologije na danem omrežju izračuna ničto in prvo Bettijevo število. To je dovolj, da zagotovimo pokritost omrežja. Ničto Bettijevo število nam pove, ali so vsi senzorji povezani med seboj, prvo pa koliko lukenj je v omrežju. Če nimamo lukenj in so vsi senzorji povezani v skupno omrežje, lahko trdimo, da je območje, ki ga raztezajo senzorji pokrito. Ripsov kompleks, ki ga zgradimo, pa v določenih primerih ne zajame topologije unije diskov pokritosti, zato je pogoj ki smo ga predstavili zadosten, ni pa tudi potreben za pokritost.

Z optimizacijo porazdelitve senzorjev se v diplomski nalogi nismo ukvarjali. Predpostavili smo, da so senzorji razporejeni vnaprej in na samo porazdelitev nimamo vpliva. V nadaljevanju bi bilo zanimivo pogledati, kako lahko s porazdelitvijo vplivamo na pokritost omrežja in na sam zajem domene, ki ga pokrivajo. Topologija je stara veda, ki je pred kratkim začela aktivno vstopati v svet računalniške tehnologije. V nalogi smo se dotaknili le njenih osnov in mislimo, da je na tem področju še veliko neodkritega potenciala.

Literatura

- [1] G. Carlsson »Topology and data«, Journal: Bull. Amer. Math. Soc. 46 (2009), 255-308.
- [2] H. Edelsbrunner, J. Harer, »Computational Topology: an Introduction«, American Mathematical Society, Providence, 2010.
- [3] L. Euler, »Solutio problematis ad geometriam situs pertinentis,« Commentarii academiae scientiarum Petropolitanae 8, str. 128-140, 1741.
- [4] R. Ghrist, »Elementary Applied Topology«, ed. 1.0, Createspace, 2014.
- [5] Helly, E. (1923), »Über Mengen konvexer Körper mit gemeinschaftlichen Punkten«, Jahresbericht der Deutschen Mathematiker-Vereinigung 32: 175–176
- [6] J. J. O'Connor, E. F. Robertson, »A history of Topology« [online], JOC/EFR maj 1996, [ogled 20.10.2016] Dostopno na: http://www-history.mcs.st-andrews.ac.uk/HistTopics/Topology_in_mathematics.html.
- [7] C. S. Raghavendra, Krishna M. Sivalingam, Taieb Znati, »Wireless Sensor Networks,« Springer US, 2004.
- [8] V. de Silva, R. Ghrist, »Coordinate-free coverage in sensor networks with controlled boundaries via homology,« Internal Journal of Robotics Research, št. 25, zv. 12, str. 1205-1222, dec. 2006.
- [9] V. de Silva, R. Ghrist, "Coverage in sensor networks via persistent homology," Alg. & Geom. Topology, 7, 339—358.
- [10] V. de Silva, R. Ghrist, »Homological sensor networks,« Notices Amer. Math. Soc., 54(1), 10-17.
- [11] D. Šoberl, (2013) »Porazdeljeno računanje homologije v brezžičnih senzorskih omrežjih«, Magistersko delo, Ljubljana: Univerza v Ljubljani, Fakulteta za računalništvo in informatiko.
- [12] L. Vietoris. "Über den höheren Zusammenhang kompakter Räume und eine Klasse von zusammenhangstreuen Abbildungen. Math. Ann. 97 (1927), 454–472.

- [13] A. J. Zomorodian, »Topology for Computing«, Cambridge university press, 2005.

